Block One or More Networks From a BGP Peer

Contents

Introduction

Prerequisites

- Requirements
- Components Used

Identifying and Filtering Routes based on NLRI

- Network Diagram

Filtering Using distribute-list with a Standard Access List

Filtering Using distribute-list with an Extended Access List

Filtering Using the ip prefix-list Command

- Filtering default routes from BGP peers

Related Information

Introduction

Route filtering is the basis by which Border Gateway Protocol (BGP) policies are set. There are number of ways to filter one or more networks from a BGP peer, including Network Layer Reachability Information (NLRI) and AS_Path and Community attributes. This document discusses filtering based on NLRI only. For information on how to filter based on AS_Path, refer to Using Regular Expressions in BGP. For additional information, refer to the BGP Filtering section of BGP Case Studies.

Prerequisites

Requirements

Cisco recommends that you have knowledge of basic BGP configuration. For more information, refer to BGP Case Studies and Configuring BGP.

Components Used

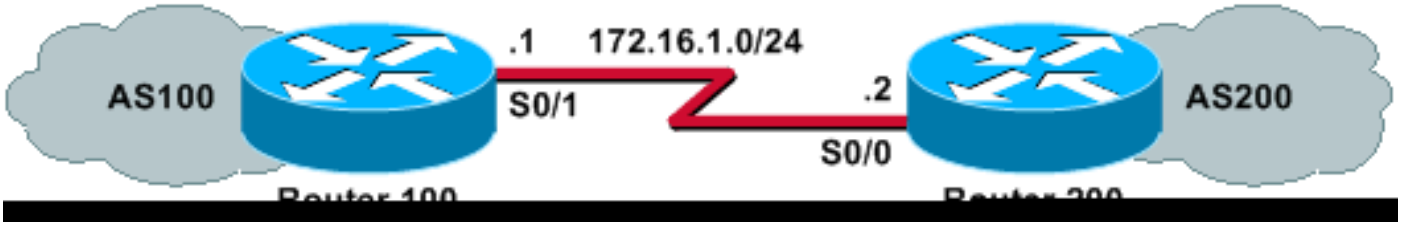
The information in this document is based on Cisco IOS® Software Release 12.2(28).

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Identifying and Filtering Routes based on NLRI

To restrict routing information that the router learns or advertises, you can use filters based on routing updates. The filters consist of an access list or a prefix list, which is applied to updates to neighbors and from neighbors. This document explores these options with this network diagram:

Network Diagram



Filtering Using distribute-list with a Standard Access List

Router 200 announces these networks to its peer Router 100:

- 192.168.10.0/24
- 10.10.10.0/24
- 10.10.0.0/19

This sample configuration enables Router 100 to deny an update for network 10.10.10.0/24 and permit the updates of networks 192.168.10.0/24 and 10.10.0.0/19 in its BGP table:

Router 100
`hostname Router 100`

```
!
router bgp 100
neighbor 172.16.1.2 remote-as 200
neighbor 172.16.1.2 distribute-list 1 in
!
access-list 1 deny 10.10.10.0 0.0.0.255
access-list 1 permit any
```

Router 200

```
hostname Router 200
!
router bgp 200
no synchronization
network 192.168.10.0
network 10.10.10.0 mask 255.255.255.0
network 10.10.0.0 mask 255.255.224.0
no auto-summary
neighbor 172.16.1.1 remote-as 100
```

This **show ip bgp** command output confirms the actions of Router 100:

<#root>

```
Router 100#
```

show ip bgp

```
BGP table version is 3, local router ID is 172.16.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 10.10.0.0/19     172.16.1.2               0             0 200 i
*> 192.168.10.0/24  172.16.1.2               0             0 200 i
```

Filtering Using distribute-list with an Extended Access List

It can be tricky to use a standard access list to filter supernets. Assume Router 200 announces these networks:

- 10.10.1.0/24 through 10.10.31.0/24
- 10.10.0.0/19 (its aggregate)

Router 100 wishes to receive only the aggregate network, 10.10.0.0/19, and to filter out all specific networks.

A standard access list, such as **access-list 1 permit 10.10.0.0 0.0.31.255**, will not work because it permits more networks than desired. The standard access list looks at the network address only and can not check the length of the network mask. That standard access-list permits the /19 aggregate as well as the more specific /24 networks.

To permit only the supernet 10.10.0.0/19, use an extended access list, such as **access-list 101 permit ip 10.10.0.0 0.0.0.0 255.255.224.0 0.0.0.0**. Refer to access-list (IP extended) for the format of the extended **access-list** command.

In our example, the source is 10.10.0.0 and the source-wildcard of 0.0.0.0 is configured for an exact match of source. A mask of 255.255.224.0, and a mask-wildcard of 0.0.0.0 is configured for an exact match of source mask. If any one of them (source or mask) does not have a exact match, the access list denies it.

This allows the extended **access-list** command to permit an exact match of source network number 10.10.0.0 with mask 255.255.224.0 (and thus, 10.10.0.0/19). The other more specific /24 networks will be filtered out.

Note: When configuring wild cards, **0** means that it is an exact match bit and **1** is a do-not-care-bit.

This is the configuration on Router 100:

Router 100

```
<#root>

hostname Router 100
  !
  router bgp 100


!--- Output suppressed.
```

```
 neighbor 172.16.1.2 remote-as 200
 neighbor 172.17.1.2 distribute-list 101 in
!
!
access-list 101 permit ip 10.10.0.0 0.0.0.0 255.255.224.0 0.0.0.0
```

The **show ip bgp** command output from Router 100 confirms that the access list is working as expected.

<#root>

```
Router 100#
```

show ip bgp

```
BGP table version is 2, local router ID is 172.16.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 10.10.0.0/19     172.16.1.2               0             0 200 i
```

As seen in this section, extended access lists are more convenient to use when some networks must be allowed and some disallowed, within the same major network. These examples provide more insight on how an extended access list can help in some situations:

- **access-list 101 permit ip 192.168.0.0 0.0.0.0 255.255.252.0 0.0.0.0**

  This access-list permits only the supernet 192.168.0.0/22.

- **access-list 102 permit ip 192.168.10.0 0.0.0.255 255.255.255.0 0.0.0.255**

  This access-list permits all the subnets of 192.168.10.0/24. In other words, it allows 192.168.10.0/24, 192.168.10.0/25, 192.168.10.128/25, and so forth: any of the 192.168.10.*x* networks with a mask that ranges from 24 to 32.

- **access-list 103 permit ip 0.0.0.0 255.255.255.255 255.255.255.0 0.0.0.255**

  This access list permits any network prefix with a mask that ranges from 24 to 32.

Filtering Using the ip prefix-list Command

Router 200 announces these networks to its peer Router 100:

- 192.168.10.0/24
- 10.10.10.0/24
- 10.10.0.0/19

The sample configurations in this section use the **ip prefix-list** command, which enables Router 100 to do two things:

- Permit updates for any network with a prefix mask length less than or equal to 19.
- Deny all network updates with a network mask length greater than 19.

Router 100

```
hostname Router 100
  !
  router bgp 100
   neighbor 172.16.1.2 remote-as 200
   neighbor 172.16.1.2 prefix-list cisco in
  !

ip prefix-list cisco seq 10 permit 0.0.0.0/0 le 19
```

Router 200

```
hostname Router 200
!
router bgp 200
no synchronization
network 192.168.10.0
network 10.10.10.0 mask 255.255.255.0
network 10.10.0.0 mask 255.255.224.0
no auto-summary
neighbor 172.16.1.1 remote-as 100
```

The **show ip bgp** command output confirms that the prefix list is working as expected on Router 100.

<#root>

```
Router 100#
```

```
show ip bgp
```

```
BGP table version is 2, local router ID is 172.16.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 10.10.0.0/19     172.16.1.2               0             0 200 i
```

In conclusion, the use of prefix lists is the most convenient way to filter networks in BGP. In some cases, however—for example, when you want to filter odd and even networks while you also control the mask length—extended access lists offers you greater flexibility and control than prefix lists.

Filtering default routes from BGP peers

You can filter or block a default route, such as 0.0.0.0/32 being advertised by the BGP peer, using the **prefix-list** command. You can see the 0.0.0.0 entry available using the **show ip bgp** command.

<#root>

```
Router 100#
```

```
show ip bgp
```

```
BGP table version is 5, local router ID is 172.16.1.1
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network          Next Hop            Metric LocPrf Weight Path
*> 0.0.0.0          172.16.1.2              0             0 200 i
```

The sample configuration in this section is performed on Router 100 using the **ip prefix-list** command.

Router 100

hostname Router 100
  !
  router bgp 100
  neighbor 172.16.1.2 remote-as 200
  neighbor 172.16.1.2 prefix-list deny-route in
  !
 	
ip prefix-list deny-route seq 5 deny 0.0.0.0/0
ip prefix-list deny-route seq 10 permit 0.0.0.0/0 le 32

If you perfom **show ip bgp** after this configuration, you will not see the 0.0.0.0 entry, which was available in the previous **show ip bgp** output.

Related Information

- **BGP Case Studies**
- **BGP Support Page**
- **Technical Support & Documentation - Cisco Systems**